

CHECK SUM CALCULATING CIRCUIT

Publication number: JP6259268

Publication date: 1994-09-16

Inventor: HIRAMATSU KYUJI

Applicant: FUJI XEROX CO LTD

Classification:

- International: G06F7/50; G06F11/10; G06F7/48; G06F11/10; (IPC1-7): G06F11/10; G06F7/50

- european:

Application number: JP19930045224 19930305

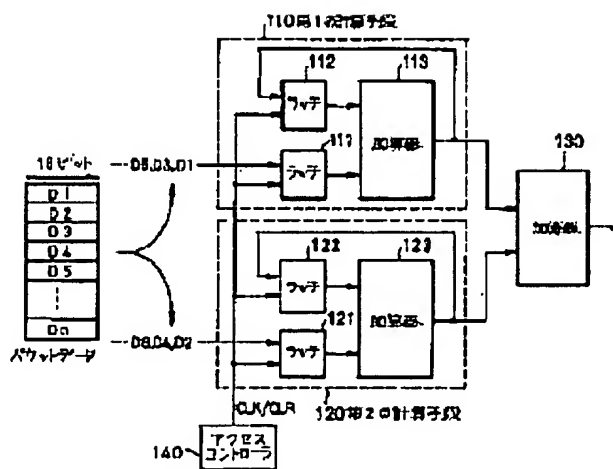
Priority number(s): JP19930045224 19930305

Report a data error here

Abstract of JP6259268

PURPOSE: To provide the check sum calculating circuit for calculating a check sum at high speed with low cost.

CONSTITUTION: A first calculating means 110 calculates the check sums of odd-numbered input data D1, D3, D5... among packet data composed of data D1, D2...Dn for the unit of 16 bits. A second calculating means 120 calculates the check sums of even-numbered input data D2, D4, D6... among those packet data. An adder 130 adds the calculated result of this first calculating means 110 and the calculated result of the second calculating means 120. This added result becomes the check sum of packet data.



Data supplied from the esp@cenet database - Worldwide

(51)IntCl. ⁵	識別記号	庁内整理番号	F 1	技術表示箇所
G 0 6 F 11/10	3 1 0 B	7313-5B		
7/50	T	9291-5B		

審査請求 未請求 請求項の数 1 O L (全 9 頁)

(21)出願番号 特願平5-45224

(22)出願日 平成5年(1993)3月5日

(71)出願人 000005496

富士ゼロックス株式会社

東京都港区赤坂三丁目3番5号

(72)発明者 平松 久二

神奈川県川崎市高津区坂戸3丁目2番1号

K S P R & D ビジネスパークビル

富士ゼロックス株式会社内

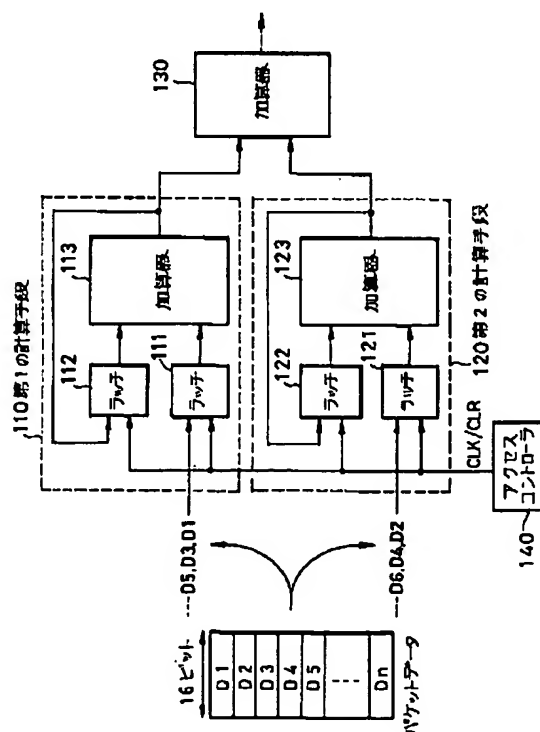
(74)代理人 弁理士 木村 高久

(54)【発明の名称】 チェックサム計算回路

(57)【要約】

【目的】低コストで、且つチェックサムの計算を高速に行うことができるチェックサム計算回路を提供する。

【構成】第1の計算手段110は、16ビット単位の手データD1、D2、…、Dnから構成されるパケットデータのうち、奇数番目に入力されるデータD1、データD3、データD5、…のチェックサムを計算する。第2の計算手段120は、上記パケットデータのうち、偶数番目に入力されるデータD2、データD4、データD6、…のチェックサムを計算する。加算器130、記第1の計算手段120による計算結果と第2の計算手段120による計算結果とを加算する。この加算結果が上記パケットデータのチェックサムとなる。



【特許請求の範囲】

【請求項1】 所定単位の前記データ毎に入力される一連の前記データのチェックサムを計算するチェックサム計算回路において、

前記一連の前記データのうち、奇数番目に入力される前記所定単位の前記データのチェックサムを計算する第1の計算手段と、

前記一連の前記データのうち、偶数番目に入力される前記所定単位の前記データのチェックサムを計算する第2の計算手段と、

前記第1の計算手段による計算結果と前記第2の計算手段による計算結果とを加算する加算手段とを具備したことを特徴とするチェックサム計算回路。

【発明の詳細な説明】

【0001】

【産業上の利用分野】 この発明は、入力されたデータのチェックサムを計算するチェックサム計算回路に関する。

【0002】

【従来の技術】 周知のように、チェックサムは、入力されたデータの誤り検出のために使用されるものであり、例えばパケットデータの誤り検出のために使用される。パケットデータの誤りの検出に使用されるチェックサムにおいては、プロトコルにより異なるものの、一般的には、パケットデータを、プロトコル毎に規定された論理式で順次計算し、その結果（チェックサム）を元に判定する方法が採用されている。

【0003】 例えばプロトコルの1つであるXNS (Xerox Network System) プロトコルでのチェックサム計算方法においては、一般に、チェックサムワード自身を除く全てのパケットデータを、16ビット単位で順次加算し、このときキャリー（桁上げ）が発生していれば、キャリーも加算し、この結果を左に1ビットローテートするようにしている。そして以上の処理を最後のパケットデータまで行い、チェックサムを求めるようにしている。

【0004】 具体的に説明すると、計算対象となるパケットデータをD1、D2、D3、D4、…とした場合、最初に「0」とデータD1とを加算する。この加算結果を左に1ビットローテートする。このローテートしたデータにデータD2を加算するが、キャリーが発生していれば、このキャリーも一緒に加算する。そして、この加算結果を左に1ビットローテートする。このローテートしたデータにデータD3を加算するが、上記同様に、キャリーが発生していれば、このキャリーも一緒に加算する。この様にXNSプロトコルでは、パケットデータをデータD1から順に1つずつ計算していく方法を採用している。

【0005】 ここで、ソフトウェアによるXNSプロトコルに従ったチェックサム計算の処理について、図4に

示すフローチャートを参照して説明する。

【0006】 ここでは、レジスタA及びレジスタBが設けられている中央処理装置（以下、CPUという）が、XNSプロトコルに従って、メモリに格納されたパケットデータのチェックサムを計算する場合について説明する。

【0007】 CPUは、レジスタBに値「0」を代入し（ステップ1）、次にメモリからデータを読み出してレジスタAに格納し（ステップ2）、更にレジスタBの値にレジスタAの値を加算する（ステップ3）。そしてこの加算によりキャリーが発生したか否かを判断する（ステップ4）。ここで、キャリーが発生した場合は、レジスタBの値をインクリメントし（ステップ5）、次に、レジスタBの値を左にローテートし（ステップ6）、更に計算が終了したか否かを判断する（ステップ7）。ステップ7において、未終了の場合には上記ステップ2に戻り、一方、終了した場合は処理を終了する。ステップ4において「NO」の場合はステップ6に進む。

【0008】 図5は従来のチェックサム計算回路のブロック図を示し、図6は図5に示したチェックサム計算回路の動作タイミングを示している。

【0009】 図5において、16ビット単位のパケットデータD1、D2、D3、D4、…、Dnは、データ転送クロックに同期して、転送データバスBUS1を経てアクセスコントローラ11及びラッチ12に入力される。例えば、図6に示すように、時点t1でデータ転送クロックに同期して転送データ（データD1）がラッチ12入力されると、ラッチ12は、時点t1でデータ転送クロックに同期しているデータラッチクロック信号CLKに同期してデータD1をラッチする。同様に、時点t2で転送データ（データD2）が、また時点tnで転送データ（データDn）がラッチ12に入力されると、時点t2でデータD2が、また時点tnでデータDnがラッチ12によってラッチされる。ラッチ12からのデータはそのままデータバスBUS2を経て16bit加算器14に入力される。

【0010】 ラッチ13は、初期状態として、アクセスコントローラ11からのクリア信号CLRによりクリアされている。このラッチ13からのデータはデータバスBUS3を経て加算器14に入力される。なおデータバスBUS3はラッチ13からのデータが1ビット左ローテートされて加算器14に入力されるようになっている。

【0011】 16bit加算器14は、ラッチ13からの16ビットのデータが1ビット左ローテートされたデータと、ラッチ12からの16ビットのデータとを加算する。この加算結果は、パケットDnについての加算処理が施されるまでは、そのままデータバスBUS4を経てラッチ13に入力される。そしてパケットデータDnについての加算処理が施されると、16bit加算器1

4から出力されるデータは、データBUS 4、BUS 5を経てバッファ15に入力される。なおデータバスBUS 5は16ビット加算器14からの加算結果が1ビット左ローテートされてバッファ15に入力されるようになっている。そしてバッファ15に保持されたデータはデータバスBUS 6に出力される。バッファ15に保持されたデータがチェックサムデータとなる。

【0012】係る構成において、チェックサム計算回路は、XNSプロトコルでの計算方法に従い、パケットデータをデータD1から順に1つずつ計算するという手順を採用している。次にこの手順に従った処理動作を、図5及び図6を参照して説明する。

【0013】最初に、時点t1(図6参照)で1回目のデータ転送クロックに同期して、データD1がアクセスコントローラ11及びラッチ12に入力されると、ラッチ12は、時点t1(図6参照)で、アクセスコントローラ11からのデータラッチクロック信号CLKの立ち上がり同期してデータD1をラッチし、このデータを加算器14へ出力する。またラッチ13からは初期設定されたデータ値「0」が加算器14に入力される。従って、加算器14からは値「0」とデータD1とが加算されたデータつまりデータD1がラッチ13に入力される。

【0014】次に、時点t2(図6参照)で2回目のデータ転送クロックに同期して、データD2がラッチ12に入力されると、ラッチ12は、時点t2(図6参照)で、上記データラッチクロック信号CLKの立ち上がり同期してデータD2をラッチし、このデータを加算器14へ出力する。一方、ラッチ13から出力されたデータ(この場合はデータD1)は1ビット左ローテートされて加算器14に入力される。加算器14は、データD1が1ビット左ローテートされたデータとラッチ12からのデータD2とを加算する。この加算結果は、そのままラッチ13に入力される。

【0015】続いて3回目のデータ転送クロックに同期して、データD3がラッチ12に入力されると、ラッチ12は、上記データラッチクロック信号CLKの立ち上がり同期してデータD3をラッチし、このデータを加算器14へ出力する。そして加算器14は、ラッチ13からの出力データが1ビット左ローテートされたデータとラッチ12からのデータD3とを加算する。この加算結果は、そのままラッチ13に入力される。

【0016】以上の動作が最後のデータDnまで繰り返される。そして加算器14によるデータDnまでの加算結果は1ビット左ローテートされたデータとしてバッファ15に保持される。このバッファ15の出力がパケットデータD1~Dnまでの全てのパケットデータのチェックサム計算結果となる。

【0017】このようなチェックサム計算回路は例えばルータ装置に設けられる。このルータ装置とは、複数の

ローカルエリアネットワーク(LAN)に接続され、これらのLANのOS I参照モデルのネットワーク層レベルでパケットを解析し、必要なパケットを最適の経路で中継するものである。このようなルータ装置は、LANを経て入力されるデータ(パケットデータ)が正しく伝送されてきたか否かを、上述したチェックサム計算回路によって判定する。

【0018】なお、ソフトウェアによりチェックサムを計算するようにしたものとしては、特開平4-125752号公報に開示されたものが知られている。

【0019】

【発明が解決しようとする課題】ところで、近年においては、上記ルータ装置に対し高速なデータ転送能力が要求されるようになって来ている。これに伴い、パケットデータのチェックサム計算を高速にする必要がある。

【0020】しかしながら、チェックサムの計算をソフトウェアで行うようにした上記従来の技術においては、多くのステップ数が必要となり、処理に時間がかかるという問題がある。

【0021】またチェックサムの計算をハードウェアで実現する場合、一般にハードウェアの中でも加算器による処理時間が、チェックサム計算処理時間中で多くの割合を占めており、チェックサム計算の処理能力は、この加算器の性能により決まるといっても過言ではない。このためチェックサム計算の処理能力を向上させるためには、高価な高速加算器を使用する必要がある。勿論、上記従来のチェックサム計算回路においても、高速化のためには高価な高速加算器を使用しなければならず、このためコスト高になってしまうという問題がある。

【0022】この発明は、低コストで、且つチェックサムの計算を高速に行うことができるチェックサム計算回路を提供することを目的とする。

【0023】

【課題を解決するための手段】この発明は、所定単位 of データ毎(図1のデータD1、D2、…、Dn)に入力される一連のデータ(図1のパケットデータ)のチェックサムを計算するチェックサム計算回路において、前記一連のデータのうち、奇数番目に入力される前記所定単位 of データ(図1のデータD1、D3、D5、…)のチェックサムを計算する第1の計算手段(図1の110)と、前記一連のデータのうち、偶数番目に入力される前記所定単位 of データ(図1のデータD2、D4、D6、…)のチェックサムを計算する第2の計算手段(図1の120)と、前記第1の計算手段による計算結果と前記第2の計算手段による計算結果とを加算する加算手段(図1の130)とを具備している。

【0024】

【作用】この発明においては、第1の計算手段が、例えばパケットデータ等の一連のデータのうち、奇数番目に入力される所定単位 of データのチェックサムを計算し、

5

6

第2の計算手段が、前記一連のデータのうち、偶数番目に入力される所定単位のデータのチェックサムを計算し、更に加算手段が、第1の計算手段及び第2の計算手段の出力を加算する。これにより、対象となるバケットデータ等の一連のデータについて、予め設定された所定単位のデータのチェックサム計算をパラレルに実行することができる。よって入力データのチェックサムを高速に計算することができる。

【0025】

【実施例】以下、本発明の実施例を添付図面を参照して 10 説明する。

【0026】図1は、本発明に係るチェックサム計算回路の原理を示す原理図である。

【0027】同図において、チェックサム計算回路は、第1の計算手段110と、第2の計算手段120と、上述した加算手段の機能を果たす加算器130とアクセスコントローラ140とを有している。

【0028】アクセスコントローラ140は、チェックサム計算回路の初期化、チェックサム計算回路へのデータサンプリング制御を行う。このデータサンプリング制御 20 においては、16ビット単位のバケットデータD1、D2、…、Dnのうち、先頭のバケットデータD1から順に数えて、奇数番目のバケットデータについては第1の計算手段110に入力されるように、また偶数番目のバケットデータについては第2の計算手段120に入力されるように、チェックサム計算データサンプルクロック信号CLK（以下、クロック信号CLKという）を、第1の計算手段110及び第2の計算手段120へ送出する。

【0029】第1の計算手段110において、ラッチ1 30 11は、クロック信号CLKに同期して奇数番目のバケットデータD1、D3、…、をラッチする。ラッチ112はアクセスコントローラ140からのクリア信号CLRに基づいてクリアされるようになっている。これらのラッチからのデータは加算器113に入力され、この加算器113によって加算された後、ラッチ112に入力される。このような処理が最後の奇数番目のバケットデータについてまで繰り返され、この最後のバケットデータについての加算器113による加算処理の結果は、ラッチ112には入力されず、加算器130に入力される。 40

【0030】一方、第2の計算手段120において、ラッチ121は、クロック信号CLKに同期して偶数番目のバケットデータD2、D4、…、をラッチする。ラッチ122はアクセスコントローラ140からのクリア信号CLRに基づいてクリアされるようになっている。これらのラッチからのデータは加算器123に入力され、この加算器123によって加算された後、ラッチ122に入力される。このような処理が最後の偶数番目のバケットデータについてまで繰り返され、この最後のバケッ 50

トデータについての加算器123による加算処理の結果は、ラッチ122には入力されず、加算器130に入力される。

【0031】そして加算器130では、加算器113からのデータとラッチ123からのデータとを加算する。この加算結果がバケットデータのチェックサム計算結果となる。

【0032】なおラッチ111、112、121、122はバケットデータの同期のために用いている。また加算器113、123、130は、図5に示した従来のチェックサム計算回路の加算器14と同一の演算速度でチェックサムを計算するものを使用している。

【0033】図2は、本発明に係るチェックサム計算回路の一実施例を示すブロック図である。

【0034】同図において、チェックサム計算回路200は、図1に示した第1の計算手段110に対応するポート210と、図1に示した第2の計算手段120に対応するポート220と、図1に示した加算器130に対応する16bit加算器230と、バッファ240と、図1に示したアクセスコントローラ140に対応するアクセスコントローラ250とを有して構成されている。

【0035】アクセスコントローラ250は、チェックサム計算回路の初期化、チェックサム計算回路へのデータサンプリング制御、セクタの切り替え制御を行うものである。

【0036】アクセスコントローラ250は、初期化の1つとして、クリア信号CLRをポート210のラッチ211、212、及びポート220のラッチ221、222へ送出し、これらのラッチをクリアする。

【0037】またアクセスコントローラ250には外部からのデータ転送クロック信号が入力されるようになっており、データ転送クロック信号に同期して、16ビット単位のバケットデータD1、D2、D3、…、Dnが順に転送データバスBUS10に出力されると、アクセスコントローラ250は、データ転送クロック信号に基づいて、このクロック信号に同期したチェックサム計算データサンプルクロック信号CLKA（以下、クロック信号CLKAという）をポート210側のラッチ211、212へ送出すると共に、上記データ転送クロック信号に同期したチェックサム計算データサンプルクロック信号CLKB（以下、クロック信号CLKB）をポート220側のラッチ221、222へ送出する。ラッチ211は、クロック信号CLKAに基づいて、データバスBUS10を経て入力される奇数番目のバケットデータをラッチし、一方、ラッチ221は、クロック信号CLKBに基づいて、データバスBUS10を経て入力される偶数番目のバケットデータをラッチする。

【0038】ポート210においては、ラッチ211のラッチデータはデータバスBUS21を経て16bit加算器213に入力され、ラッチ212のラッチデータ

はデータバスBUS 22を経て16bit加算器213に入力される。

【0039】データバスBUS 21は、ラッチ211のラッチデータがそのまま16bit加算器213に入力されるようになっており、またデータバスBUS 22は、ラッチ212のラッチデータが2ビット左ローテートされて16bit加算器213に入力されるようになっている。

【0040】16bit加算器213は、データバスBUS 21を経て入力された16ビットのデータと、データバスBUS 22を経て入力された16ビットのデータとを加算する。なお16bit加算器213は、加算処理においてキャリーが発生している場合はキャリーも加算する。16bit加算器213による加算結果は、データバスBUS 23を経てラッチ212及びセクタ214に入力される。

【0041】ラッチ212では、入力されるクロック信号CLKAに基づいて16bit加算器213からの加算結果をラッチする。

【0042】一方、セクタ214には、データBUS 23からの16ビットのデータと、データバスBUS 24からの16ビットのデータとが入力される。なおデータバスBUS 24は、16bit加算器213による加算結果が1ビット左ローテートされてセクタ214に入力されるようになっている。セクタ214は、アクセスコントローラ250からのセレクト信号に基づいて、データバスBUS 23からのデータ或いはデータバスBUS 24からのデータいずれかを選択して出力する。

【0043】なおポート220も上記ポート210と同様の構成になっている。

【0044】ここで、アクセスコントローラ250から出力されるセレクト信号は、最後のパケットデータDnが、ポート210側で処理される場合とポート220側で処理される場合とでは異なっている。

【0045】すなわち、データDnがポート210で処理された場合、アクセスコントローラ250は、セクタ214に対してはデータバスBUS 23からのデータを、一方、セクタ224に対してはデータバスBUS 33からのデータを選択して出力するように、セレクト信号を各セクタへ送出する。このとき、セクタ214からは16bit加算器213による加算結果(16ビットのデータ)がそのまま出力され、セクタ224からは16bit加算器213による加算結果が1ビット左ローテートされた16ビットのデータが出力される。

【0046】一方、データDnがポート220で処理された場合、アクセスコントローラ250は、セクタ214に対してはデータバスBUS 24からのデータを、一方、セクタ224に対してはデータバスBUS 34

からのデータを選択して出力するように、セレクト信号を各セクタへ送出する。このとき、セクタ214からは16bit加算器213による加算結果が1ビット左ローテートされた16ビットのデータが出力され、一方、セクタ224からは16bit加算器213による加算結果(16ビットのデータ)がそのまま出力される。

【0047】なおアクセスコントローラ250は、最後のデータDnがポート210側で処理されるのか、或いはポート220側で処理されるのかを、次のようにして認識する。

【0048】上記XNSのパケットデータには、データ長を示すレングスが格納されているので、アクセスコントローラ250は、そのレングス値とデータ転送クロック数とに基づいて、データDnがどちらのポートにて処理されるかを認識する。上記パケットデータに格納されているレングスはデータ長をバイト単位で表したものであるため、レングス(L)が奇数の場合はレングス値に値「1」を加算して偶数に変換する(つまりレングス $(L \leftarrow L + 1)$)。なおレングス(L)が偶数の場合は上述した変換処理は不要となる。レングス(L)が奇数か偶数かはレングスのLBS(最下位ビット)を参照することにより認識することができ、このLBS(最下位ビット)が、1のときは奇数、0のときは偶数である。

【0049】そして最初のデータD1が入力されてからのデータ転送クロック数をカウントし、このカウント値 $\times 2$ (1クロックで2バイトのデータが転送されるため)の値が変換されたレングス値と一致した場合は、データDnの処理が終了したものと判断し、各セクタに対してセレクト信号を出力する。ここでデータDnは、カウンタ値が奇数の場合にはポート210側で処理されることになり、カウンタ値が偶数の場合はポート220側で処理されることになる。

【0050】例えばレングス値が512バイトでカウンタ値が256回のときは、レングス値=512バイトと、カウンタ値 $\times 2 = 512$ バイトとが一致する。このときは、カウンタ値が偶数であるので、最後のデータDnはポート220側で処理されることとなる。

【0051】またレングス値が509バイトでカウンタ値が255回のときは、レングス値509バイトが奇数のため、この値を、レングス値 $= 509 + 1 = 510$ バイトに変換する。この値510バイトと、カウンタ値 $\times 2 = 255 \times 2$ バイト $= 510$ バイトとが一致する。このときは、カウンタ値が奇数であるので、最後のデータDnはポート210側で処理されることとなる。

【0052】そして16bit加算器230は、データバスBUS 25を経て入力されるセクタ214からの16ビットのデータと、データバスBUS 35を経て入力されるセクタ224からの16ビットのデータとを加算し、この加算結果を出力する。なお16bit加算

器230は、加算処理においてキャリーが発生している場合はキャリーも加算する。

【0053】16bit加算器230による加算結果はデータバスBUS40を経てバッファ240に入力され、更にデータバスBUS50に出力される。データバスBUS40は、16bit加算器213からの加算結果が1ビット左ローテートされてバッファ240に入力されるようになっている。従って、バッファ240には16bit加算器230による加算結果が1ビット左ローテートされた値が保持されることになる。このデータ

が、XNSプロトコルにおけるチェックサムとなる。

【0054】図3は、本実施例のチェックサム計算回路の動作を示すタイミングチャートである。なおこの実施例においては、16bit加算器213、223、230はいずれも、図5に示した従来のチェックサム計算回路の16ビット加算器と同一の演算速度でチェックサムを計算するものを使用している。

【0055】次に、チェックサム計算回路のチェックサム計算処理について、図2及び図3を参照して説明する。

【0056】1：最初に時点t1（図3参照）でデータ転送クロック（1回目）（図3（a）参照）によりパケットデータD1（図3（b）参照）が入力されたとする。するとポート210においては、ラッチ211は、時点t1（図3（c）参照）でクロック信号CLKAの立上りに同期してデータD1をラッチする。ラッチされたデータD1は、データバスBUS21を経て16bit

加算器213に入力される。ラッチ212はクリア信号CLR信号により事前リセットされているので、ラッチ212の出力は“0”である。従って16bit加算器213からは、データD1がそのまま出力され、データバスBUS23を経てラッチ212に入力される。

【0057】2：次に時点t2（図3参照）でデータ転送クロック（2回目）（図3（a）参照）によりパケットデータD2（図3（b）参照）が入力される。するとポート220においては、ラッチ221は、時点t2（図3（d）参照）でクロック信号CLKBの立上りに同期してデータD2をラッチする。ラッチされたデータD2は、データバスBUS31を経て16bit加算器223に入力される。ラッチ222はクリア信号CLR

により事前リセットされているので、ラッチ221の出力は“0”である。したがって16bit加算器223からは、データD2がそのまま出力され、データバスBUS34を経てラッチ222に入力される。

【0058】3：続いて時点t3（図3参照）でデータ転送クロック（3回目）（図3（a）参照）によりパケットデータD3（図3（b）参照）が入力される。するとポート210においては、時点t3（図3（b）参照）でクロック信号CLKAの立上りに同期して、ラッチ211はデータD3をラッチし、一方、ラッチ212

は16bit加算器213からの加算結果（つまりデータD1）をラッチする。そして16bit加算器213は、ラッチ211からのデータD3と、ラッチ211からのデータ（つまりデータD1）が2ビット左ローテートされたデータとを加算する。この加算結果はラッチ212に入力される。

【0059】4：更に、次のデータ転送クロック（4回目）によりパケットデータD4が入力される。するとポート220においては、クロック信号CLKBの立上りに同期して、ラッチ221はデータ4をラッチし、一方、ラッチ222は16bit加算器223からの加算結果（つまりデータD2）をラッチする。そして16bit加算器223は、ラッチ221からのデータD4と、ラッチ222からのデータ（つまりデータD2）が2ビット左ローテートされたデータとを加算する。この加算結果はラッチ222に入力される。

【0060】以上のような処理を、最後のデータDnについて処理するまで、ポート210においては上記（3）の処理を繰り返し、ポート220においては上記（4）の処理を繰り返す。

【0061】そして最後のデータDnがポート210側で処理されたとすると、セクタ214は、アクセスコントローラ250からのセレクト信号に基づいて、データバスBUS23を経て入力されたデータを選択して、16bit加算器230へ出力する。またセクタ224は、アクセスコントローラ250からのセレクト信号に基づいて、データバスBUS33を経て入力されたデータを選択して、16bit加算器230へ出力する。

【0062】16bit加算器230では、セクタ214からのデータとセクタ224からのデータとを加算する。この加算結果は1ビット左ローテートされてバッファ240に保持される。このバッファ240の出力がチェックサム計算結果となる。

【0063】以上説明したように本実施例によれば、従来と比較して、従来の加算器と同一の加算器を使用した場合、すなわち従来のチェックサム計算回路の加算器14の所要計算時間を、図6（a）に示すように時間tとし、また本実施例のチェックサム計算回路の加算器213、223、230の所要計算時間も共に図3（a）に示すように時間tとした場合は、図3及び図6からも明らかに、2倍の速度で転送データをサンプリングすることができる。

【0064】また従来の回路全体の計算時間をT、本実施例の加算器230の計算時間を上記時間tとすると、本実施例の回路全体の計算時間は、 $(T/2) + t$ となり、従来と比較して、大幅に計算時間を短縮することができる。

【0065】また従来においては、計算時間を短縮するためには、高価な高速加算器が必要であるが、この実施例においては、安価な低速加算器で同程度の計算時間を

達成することができる。

【0066】なお本実施例のチェックサム計算回路は、ルータ装置や、ネットワークに接続されたネットワーク資源とのデータ送受を行う装置例えばネットワーク接続装置や、データを中継する装置などに組み込むことができる。従って、例えばルータ装置に本実施例のチェックサム計算回路を設けることにより、このルータ装置においては高速なチェックサム計算処理を行うことが可能となり、結果として、ルータ装置のデータ転送能力を向上させることができる。

【0067】

【発明の効果】以上説明したように本発明によれば、第1の計算手段が、一連のデータのうち、奇数番目に入力される所定単位のデータのチェックサムを計算し、第2の計算手段が、前記一連のデータのうち、偶数番目に入力される所定単位のデータのチェックサムを計算し、更に加算手段が、第1の計算手段及び第2の計算手段の出力を加算するようにしているのので、一連のデータについて、予め設定された所定単位のデータのチェックサム計算をパラレルに実行することができる。従って、前記所定単位のデータを高速でサンプリングすることができることとなり、よってチェックサム計算処理を迅速に行うことができるという利点がある。

【図面の簡単な説明】

【図1】本発明に係るチェックサム計算回路の原理を示す原理図。

【図2】本発明に係るチェックサム計算回路の一実施例を示すブロック図。

【図3】図2に示した実施例の処理動作を示すタイミングチャート。

【図4】従来におけるソフトウェアによるチェックサム計算の処理動作を示すフローチャート。

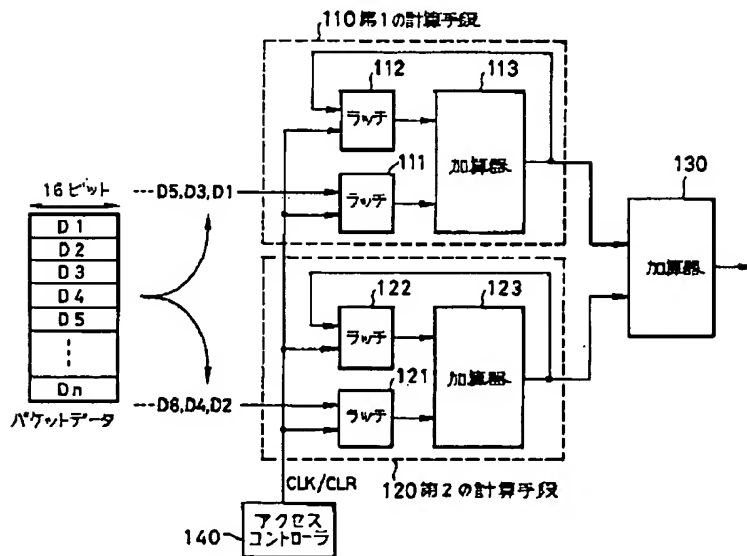
10 【図5】従来のチェックサム計算回路を示すブロック図。

【図6】図5に示したチェックサム計算回路の処理動作を示すタイミングチャート。

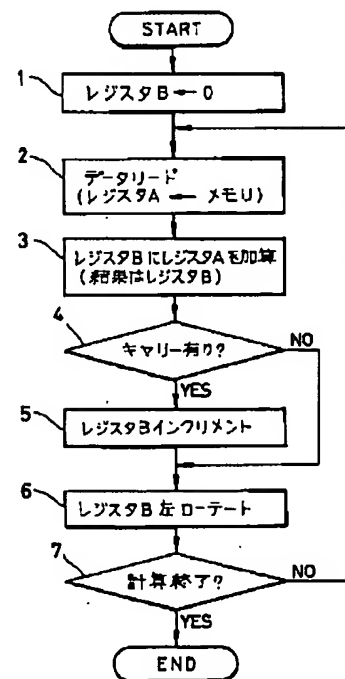
【符号の説明】

110…第1の計算手段、120…第2の計算手段、112、113、122、123…ラッチ、111、121…加算器、130…加算器、140、250…アクセスコントローラ、210、220…ポート、213、223、230…16bit加算器、214、224…セクタ、240…バッファ、BUS10、BUS21～BUS25、BUS31～BUS35、BUS40、BUS50…データバス。

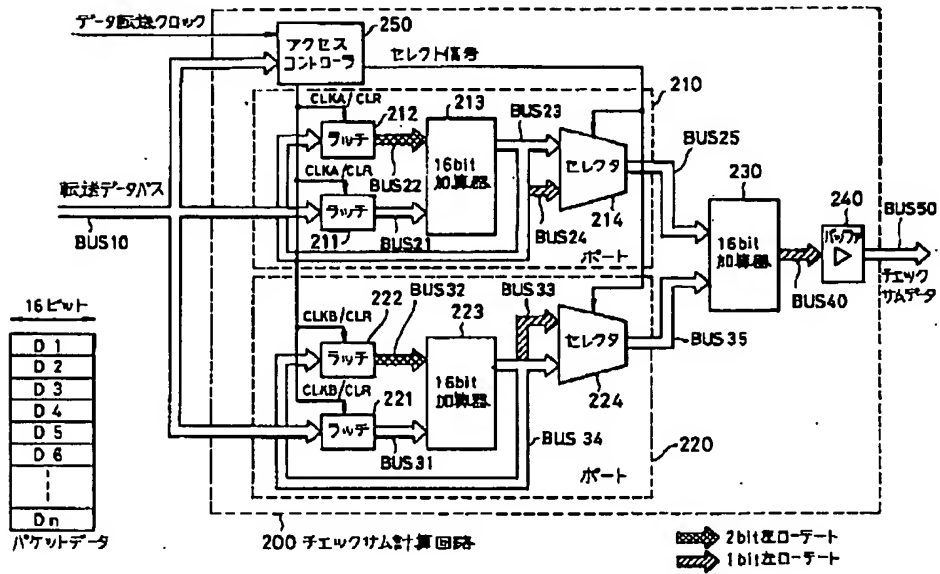
【図1】



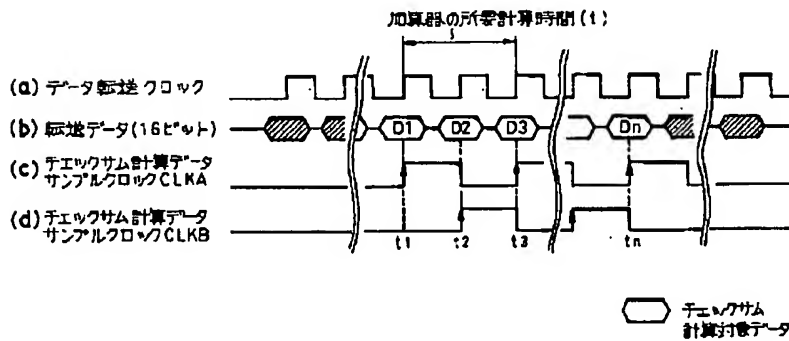
【図4】



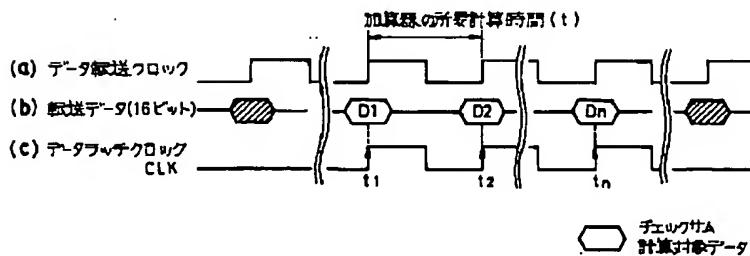
【図2】



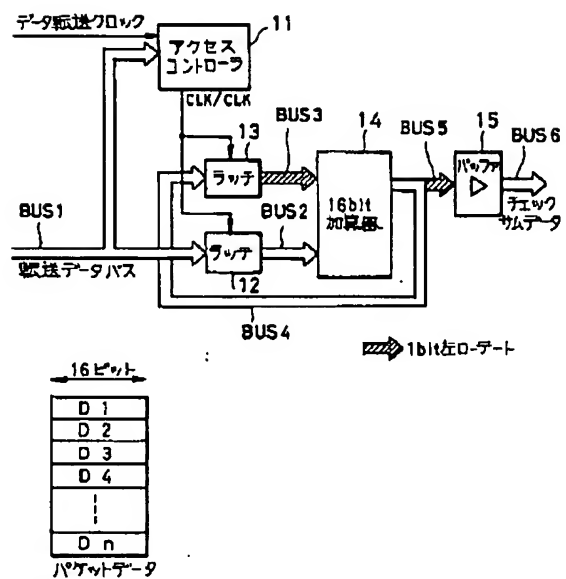
【図3】



【図6】



【図5】



**This Page is Inserted by IFW Indexing and Scanning
Operations and is not part of the Official Record.**

BEST AVAILABLE IMAGES

Defective images within this document are accurate representations of the original documents submitted by the applicant.

Defects in the images include but are not limited to the items checked:

- ☐ **BLACK BORDERS**
- ☐ **IMAGE CUT OFF AT TOP, BOTTOM OR SIDES**
- ☒ **FADED TEXT OR DRAWING**
- ☒ **BLURRED OR ILLEGIBLE TEXT OR DRAWING**
- ☐ **SKEWED/SLANTED IMAGES**
- ☐ **COLOR OR BLACK AND WHITE PHOTOGRAPHS**
- ☐ **GRAY SCALE DOCUMENTS**
- ☒ **LINES OR MARKS ON ORIGINAL DOCUMENT**
- ☒ **REFERENCE(S) OR EXHIBIT(S) SUBMITTED ARE POOR QUALITY**
- ☐ **OTHER:** _____

IMAGES ARE BEST AVAILABLE COPY.

As rescanning these documents will not correct the image problems checked, please do not report these problems to the IFW Image Problem Mailbox.